

[First Hit](#)   [Fwd Refs](#)☐ [Generate Collection](#) [Print](#)

L14: Entry 12 of 67

File: USPT

Apr 9, 2002

DOCUMENT-IDENTIFIER: US 6370634 B1

TITLE: Data flow computer with two switches

Brief Summary Text (7):

In this system Von Neumann's principle of data processing is used. Every central processor contains a conforming (conjugating) unit, a block for performing procedures, an indexing block, a block for value retrieving, a block for processing strings, an arithmetic-logical unit, a block of the basic registers, a unit for instructions forming, a control unit, a unit for the distribution of stack addresses, a buffering stack of operands, an associative memory unit, a unit for transformation of the mathematical addresses into the physical ones, a block of memory for buffering instructions, a block for analysis of interrupts.

## Refine Search

### Search Results -

Terms	Documents
((conjugat\$ or conform\$) near3 processo\$)	480

Database:

US Pre-Grant Publication Full-Text Database  
 US Patents Full-Text Database  
 US OCR Full-Text Database  
 EPO Abstracts Database  
 JPO Abstracts Database  
 Derwent World Patents Index  
 IBM Technical Disclosure Bulletins

Search:

L15

Refine Search

Recall Text

Clear

Interrupt

### Search History

 DATE: Wednesday, June 30, 2004    [Printable Copy](#)    [Create Case](#)

<u>Set Name</u> side by side	<u>Query</u>	<u>Hit Count</u>	<u>Set Name</u> result set
	<i>DB=USPT; PLUR=YES; OP=ADJ</i>		
<u>L15</u>	((conjugat\$ or conform\$) near3 processo\$)	480	<u>L15</u>
<u>L14</u>	(conjugat\$ near3 processo\$)	67	<u>L14</u>
<u>L13</u>	6516022.pn.	1	<u>L13</u>
<u>L12</u>	l6 and conjug\$	0	<u>L12</u>
<u>L11</u>	l6 and (compress\$ or reduc\$ or loss\$)	1	<u>L11</u>
<u>L10</u>	l6 and graphical near4 (Boat\$)	1	<u>L10</u>
<u>L9</u>	l6 and (non-affine)	1	<u>L9</u>
<u>L8</u>	L6 and (instruction\$ Or process\$) near9 (dimension\$)	1	<u>L8</u>
<u>L7</u>	L6 and (instruction\$ Or process\$)	1	<u>L7</u>
<u>L6</u>	6078745.pn.	1	<u>L6</u>
<u>L5</u>	(convert\$ or map\$ or transfer\$ or transform\$) near6 (multi-dimension\$ near8 one-dimension\$)	44	<u>L5</u>
<u>L4</u>	L3 and architect\$	42	<u>L4</u>

L3 L2 and map\$  
L2 L1 and (processor\$ or execut\$ or instruction\$)  
L1 (multi-dimension\$ near8 one-dimension\$)

92 L3  
185 L2  
258 L1

END OF SEARCH HISTORY



US Patent &amp; Trademark Office

[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide

edit and distance and compression

SEARCH

THE ACM DIGITAL LIBRARY


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)
Terms used [edit](#) and [distance](#) and [compression](#)

Found 9,753 of 138,663

Sort results by

relevance

☒ Save results to a BinderTry an [Advanced Search](#)Try this search in [The ACM Guide](#)

Display results

expanded form

☒ Search Tips☐ Open results in a new window

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale ☐ ☐ ☐ ☐ ☐1 [The string edit distance matching problem with moves](#)

Graham Cormode, S. Muthukrishnan

January 2002 **Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms**Full text available: [pdf\(1.13 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

The edit distance between two strings  $S$  and  $R$  is defined to be the minimum number of character inserts, deletes and changes needed to convert  $R$  to  $S$ . Given a text string  $t$  of length  $n$ , and a pattern string  $p$  of length  $m$ , informally, the string edit distance matching problem is to compute the smallest edit distance between  $p$  and substrings of  $t$ . A well known dynamic programming algorithm takes time  $O(nm)$  to solve ...

2 [Session 12C: The similarity metric](#)

Ming Li, Xin Chen, Xin Li, Bin Ma, Paul Vitányi

January 2003 **Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms**Full text available: [pdf\(1.04 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

A new class of metrics appropriate for measuring effective similarity relations between sequences, say one type of similarity per metric, is studied. We propose a new "normalized information distance", based on the noncomputable notion of Kolmogorov complexity, and show that it minorizes every metric in the class (that is, it is universal in that it discovers all effective similarities). We demonstrate that it too is a metric and takes values in  $[0, 1]$ ; hence it may be called the *similarity  $m$*  ...

3 [A guided tour to approximate string matching](#)

Gonzalo Navarro

March 2001 **ACM Computing Surveys (CSUR)**, Volume 33 Issue 1Full text available: [pdf\(1.19 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

We survey the current techniques to cope with the problem of string matching that allows errors. This is becoming a more and more relevant issue for many fast growing areas such as information retrieval and computational biology. We focus on online searching and mostly on edit distance, explaining the problem and its relevance, its statistical behavior, its history and current developments, and the central ideas of the algorithms and their complexities. We present a number of experiments to ...

**Keywords:** Levenshtein distance, edit distance, online string matching, text searching